UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/755,449	01/13/2004	Vladimir Vasekin	550-500	1088

23117 7590 09/29/2006

NIXON & VANDERHYE, PC
901 NORTH GLEBE ROAD, 11TH FLOOR
ARLINGTON, VA 22203

EXAMINER	
GEIB, BENJAMIN P	

ART UNIT	PAPER NUMBER
2181	

DATE MAILED: 09/29/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/755,449	VASEKIN, VLADIMIR	
	Examiner	Art Unit	
	Benjamin P. Geib	2181	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *13 July 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-4,9-14,19-24,29 and 30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-4,9-14,19-24,29 and 30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

FRITZ FLEMING
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100
9/17/2006

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-4, 9-14, 19-24, 29, and 30 have been examined.

2. It is hereby acknowledged that the following papers have been received and placed of record in the file: Amendment as received on 07/13/2006.

Withdrawn Rejections

3. Applicant, via amendment, has overcome the 35 U.S.C. § 112, second paragraph, rejections set forth in the previous Office Action. Consequently, these rejections have been withdrawn by the examiner.

4. Applicant, via amendment, has overcome the 35 U.S.C. § 101 rejections set forth in the previous Office Action. Consequently, these rejections have been withdrawn by the examiner.

Claim Rejections - 35 USC § 112

5. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

6. Claims 9, 19, and 29 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

7. Claims 9, 19, and 29 recite the limitation "said address of said execute block instruction". Since there is no previous mention of an "address of said execute block instruction", there is insufficient antecedent basis for this limitation in the claims. Since there is a previous mention of "an address indicative of a

memory location of said execute block instruction" (in claims 1, 11, and 21, respectively), the limitation "said address of said execute block instruction" will be interpreted as "said address indicative of a memory location of said execute block instruction" for the remainder of the examination.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

9. Claims 1-4, 9-14, 19-24, 29, and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Fraser, U.S. Patent No. 6,907,598.

10. Referring to claim 1, Fraser has taught an apparatus for processing data, said apparatus comprising:

an instruction fetching circuit *[access module; Fig. 3, component 332]* operable to fetch program instructions from a sequence of memory locations *[column 11, line 65 – column 12, line 3]*;

an instruction decoder *[evaluate module; Fig. 3, component 334]* responsive to program instructions fetched by said instruction fetching circuit to control data processing operations specified by said program instructions *[column 12, lines 3-13]*; and

an execution circuit *[execute module; Fig. 3, component 338]* operable under control of said instruction decoder to execute said data processing operations *[column 12, lines 7-13]*;

a program counter register operable when said apparatus is executing said program instructions from said sequence of memory locations to store an address indicative of a memory location of a program instruction being executed within said program instructions from said sequence of memory locations *[The program counter stores a pointer indicating the next instruction to be executed; column 12, lines 5-7]*; and

a block counter register operable to store a block count value indicative of a location of a program instruction being executed within a block of two or more program instructions *[The count module stores the length parameter (i.e. block count value) of an echo instruction to determine the remaining instructions in the set of instructions (i.e. location of a program instruction being executed within the block); column 12, lines 32-40]*,

wherein said instruction decoder is responsive to an execute block instruction *[echo instruction]* to trigger fetching of a block of two or more program instructions *[set of instructions]* by said instruction fetching circuit and execution of said block of two or more program instructions by said execution circuit, said block of two or more instructions containing a number of program instructions specified by a block length field *[length parameter]* within said executed block instruction and being stored at a memory location specified by a location field *[displacement parameter]* within said execute block instruction *[When the*

evaluate module determines there is an echo instruction it triggers the fetching and execution of a set of instructions, whose number and location are indicated by the length and displacement parameters, respectively, of the echo instruction; column 12, lines 14-44]; and

wherein when executing said block of two or more program instructions, said program counter register is configured to store an address indicative of a memory location of said execute block instruction *[the address of an instruction within the set of instructions]* and said block counter register is configured to store a block count value indicative of said program instruction location of a program instruction being executed with said block of two or more program instructions *[the count of remaining instruction in the set of instruction]* corresponding to said execute block instruction *[column 12, lines 14-37]*,

Fraser has not explicitly taught an exception handling circuit operable upon occurrence of an exception during execution of said block of two or more instructions to store said block count value, and upon completion of handling of said exception, to restart execution of said block of two or more program instructions at a program instruction within said block of two or more instructions indicated by said block count value.

However, Examiner takes Official Notice that exception handling circuits that save a processor's execution state upon occurrence of an exception and restart execution at the program instruction indicated by the saved execution state upon completion of handling the exception are conventional and well-known.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to included an exception handling circuit that saves the processor's execution state, which in the instant case includes a block count value, upon occurrence of an exception and restarts execution at the program instruction indicated by the saved state, including the block count value, upon completion of handling the exception since doing so allows the processor to properly resume execution of a program after an exception.

11. Referring to claim 2, Fraser has taught the apparatus as claimed in claim 1, wherein after execution of said block of two or more program instructions a return is made to a program instruction outside of said block of two or more program instructions *[column 12, lines 32-44]*.

12. Referring to claim 3, Fraser has taught the apparatus as claimed in claim 1, wherein said return is to a program instruction immediately following said execute block instruction within said sequence of memory locations *[column 12, lines 32-44]*.

13. Referring to claim 4, Fraser has taught the apparatus as claimed in claim 1, wherein said location field is an offset field *[displacement parameter]* specifying said location of said block of two or more program instructions relative to a memory location of said execute block instruction *[The displacement parameter of an echo instruction specifies the location of the set of instructions relative to the echo instruction; column 12, lines 20-30]*.

14. Referring to claim 9, Fraser has taught the apparatus as claimed in claim 1.

Fraser has not explicitly taught that the exception handling circuit is operable to store said address indicative of a memory location of said execute block instruction (i.e. the program counter) upon occurrence of an exception and to restore said address indicative of a memory location of said execute block instruction to said program counter register upon said completion of handling of said exception.

However, Examiner takes Official Notice that exception handling circuits that save a processor's execution state upon occurrence of an exception and restore the saved execution state upon completion of handling the exception are conventional and well-known.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the exception handling circuit to save the program counter (i.e. address indicative of a memory location of said execute block instruction), which is part of the processor's execution state, upon occurrence of an exception and restores the saved state, including the program counter, upon completion of handling the exception since doing so allows the processor to properly resume execution of a program after an exception.

15. Referring to claim 10, Fraser has taught the apparatus as claimed in claim 2, wherein upon completion of execution of said block of two or more program instructions said instruction decoder is operable to return processing to a program instruction following said execute block instruction as indicated by said program counter register *[Upon completion of the set of instructions the program counter is restored and the processing is restored to a program instruction*

following the echo instruction as indicated by the program counter; column 12, lines 37-44].

16. Referring to claim 11, Fraser has taught a method for processing data, said method comprising the steps of:

fetching program instructions from a sequence of memory locations with an instruction fetching circuit *[access module; Fig. 3, component 332; column 11, line 65 – column 12, line 3]*;

controlling data processing operations specified by said program instructions with an instruction decoder *[evaluate module; Fig. 3, component 334; column 12, lines 3-13]*; and

executing said data processing operations with an execution circuit controlled by said instruction decoder *[execute module; Fig. 3, component 338; column 12, lines 7-13]*;

storing within a program counter register an address indicative of a memory location of a program instruction being executed within said sequence of program instructions *[The program counter stores a pointer indicating the next instruction to be executed; column 12, lines 5-7]*; and

storing within a block counter register a block count value indicative of a location of a program instruction being executed within a block of two or more program instructions *[The count module stores the length parameter (i.e. block count value) of an echo instruction to determine the remaining instructions in the set of instructions (i.e. location of a program instruction being executed within the block); column 12, lines 32-40]*,

wherein said instruction decoder is responsive to an execute block instruction *[echo instruction]* to trigger fetching of a block of two or more program instructions *[set of instructions]* by said instruction fetching circuit and execution of said block of two or more program instructions by said execution circuit, said block of two or more instructions containing a number of program instructions specified by a block length field *[length parameter]* within said executed block instruction and being stored at a memory location specified by a location field *[displacement parameter]* within said execute block instruction *[When the evaluate module determines there is an echo instruction it triggers the fetching and execution of a set of instructions, whose number and location are indicated by the length and displacement parameters, respectively, of the echo instruction; column 12, lines 14-44]*,

wherein when executing said block of two or more program instructions, said program counter register stores an address indicative of a memory location of said execute block instruction *[the address of an instruction within the set of instructions]* and said block counter register stores a block count value indicative of said program instruction location of a program instruction being executed within said block of two or more program instructions *[the count of remaining instruction in the set of instruction]* corresponding to said execute block instruction *[column 12, lines 14-37]*

Fraser has not explicitly taught upon occurrence of an exception during execution of said block of two or more instructions, storing said block count value and upon completion of handling of said exception, restarting execution of said

block of two or more program instructions at a program instruction within said block of two or more instructions indicated by said block count value.

However, Examiner takes Official Notice that saving a processor's execution state upon occurrence of an exception and restarting execution at the program instruction indicated by the saved execution state upon completion of handling the exception is conventional and well-known.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the method of Fraser to include saving the processor's execution state, which in the instant case includes a block count value, upon occurrence of an exception and restarting execution at the program instruction indicated by the saved state, including the block count value, upon completion of handling the exception since doing so allows the processor to properly resume execution of a program after an exception.

17. Referring to claim 12, given the similarities between claim 2 and claim 12 the arguments as stated for the rejection of claim 2 also apply to claim 12.

18. Referring to claim 13, given the similarities between claim 3 and claim 13 the arguments as stated for the rejection of claim 3 also apply to claim 13.

19. Referring to claim 14, given the similarities between claim 4 and claim 14 the arguments as stated for the rejection of claim 4 also apply to claim 14.

20. Referring to claim 19, given the similarities between claim 9 and claim 19 the arguments as stated for the rejection of claim 9 also apply to claim 19.

21. Referring to claim 20, given the similarities between claim 10 and claim 20 the arguments as stated for the rejection of claim 10 also apply to claim 20.

22. Referring to claim 21, Fraser has taught a computer program product stored on a computer-readable storage medium including a computer program operable to control a data processing apparatus having

an instruction fetching circuit *[access module; Fig. 3, component 332]* operable to fetch program instructions from a sequence of memory locations *[column 11, line 65 – column 12, line 3]*,

an instruction decoder *[evaluate module; Fig. 3, component 334]* responsive to program instructions fetched by said instruction fetching circuit to control data processing operations specified by said program instructions *[column 12, lines 3-13]*, and

an execution circuit *[execute and Echo modules; Fig. 3, components 338 & 336]* operable under control of said instruction decoder to execute said data processing operations *[column 12, lines 7-13]* and to store within a program counter register an address indicative of a memory location of a program instruction being executed within said sequence of program instructions *[The program counter stores a pointer indicating the next instruction to be executed; column 12, lines 5-7]*;

wherein said computer program including one or more an execute block instructions *[echo instruction]* operable to trigger fetching of a block of two or more program instructions *[set of instructions]* by said instruction fetching circuit and execution of said block of two or more program instructions by said execution circuit and storing within a block counter register a block count value indicative of a location of a program

instruction being executed within said block of two or more program instructions *[The count module stores the length parameter (i.e. block count value) of an echo instruction to determine the remaining instructions in the set of instructions (i.e. location of a program instruction being executed within the block); column 12, lines 32-40]*, said block of two or more instructions containing a number of program instructions specified by a block length field *[length parameter]* within said executed block instruction and being stored at a memory location specified by a location field *[displacement parameter]* within said execute block instruction *[When the evaluate module determines there is an echo instruction it triggers the fetching and execution of a set of instructions, whose number and location are indicated by the length and displacement parameters, respectively, of the echo instruction; column 12, lines 14-44]*, and

wherein said one or more execute block instructions is operable to cause, during execution of said block of two or more program instructions, said program counter register to store an address indicative of a memory location of said execute block instruction *[the address of an instruction within the set of instructions]* and to cause said block counter register to store a block count value indicative of said program instruction location of a program instruction being executed within said block of two or more program instructions *[the count of remaining instruction in the set of instruction]* corresponding to said execute block instruction *[column 12, lines 14-37]*; and

Fraser has not explicitly taught exception handling code, operable upon occurrence of an exception during execution of said block of two or more instructions, storing said block count value and upon completion of handling of said exception, restarting execution of said block of two or more program instructions at a program instruction within said block of two or more instructions indicated by said block count value.

However, Examiner takes Official Notice that exception handling code, operable to save a processor's execution state upon occurrence of an exception and restart execution at the program instruction indicated by the saved execution state upon completion of handling the exception is conventional and well-known.

It would have been obvious to one of ordinary skill in the art at the time the invention was made to modify the system of Fraser to include exception handling code, operable to save the processor's execution state, which in the instant case includes a block count value, upon occurrence of an exception and restart execution at the program instruction indicated by the saved state, including the block count value, upon completion of handling the exception since doing so allows the processor to properly resume execution of a program after an exception.

23. Referring to claim 22, given the similarities between claim 2 and claim 22 the arguments as stated for the rejection of claim 2 also apply to claim 22.

24. Referring to claim 23, given the similarities between claim 3 and claim 23 the arguments as stated for the rejection of claim 3 also apply to claim 23.

25. Referring to claim 24, given the similarities between claim 4 and claim 24 the arguments as stated for the rejection of claim 4 also apply to claim 24.

26. Referring to claim 29, given the similarities between claim 9 and claim 29 the arguments as stated for the rejection of claim 9 also apply to claim 29.

27. Referring to claim 30, given the similarities between claim 10 and claim 30 the arguments as stated for the rejection of claim 10 also apply to claim 30.

Response to Arguments

28. Applicants arguments filed on July 13, 2006, have been fully considered but they are not found persuasive.

29. In response to Applicant's request for the Examiner to supply a prior art reference to support the Official Notice position regarding exception handling, the Examiner cites Hennessy et al., "Computer Architecture: A Quantitative Approach" pages A-42 to A-44. The cited section describes stopping and restarting execution for the purpose of exception handling. In particular, Hennessy describes that "the pipeline must be safely shut down and the state saved so that the instruction can be restarted in the correct state" (1st paragraph of section label "Stopping and Restarting Execution" on page A-42). Hennessy further details that this state typically is comprised of the program counter (PC) of the instruction at which to restart (1st paragraph of section label "Stopping and Restarting Execution" on page A-42). The Examiner notes that it is this concept of exception handling and stopping/restarting execution by saving the execution

state that is being applied to the system of Fraser. As noted above, when this concept of saving the state so that the instruction can be restarted in the correct state is applied to the system of Fraser, the program counter and block counter register are saved and restored in the manner described in the above rejection.

30. Applicant argues the novelty/rejection of claim 1 on page 12 of the remarks, in substance that:

> "Another difference from Fraser's Echo instruction is the claimed execute block instruction encodes the size of the block – the claimed block length field – rather than the number of instructions" (2nd paragraph on page 12)

These arguments are not found persuasive for the following reasons:

The claim language regarding the claimed block length field as written (i.e. "a number of program instructions specified by a block length field with said executed block instruction") only requires that block length field specify a number of program instructions within the block of two or more instructions. The length parameter of the Echo instruction species the number of instructions in the block of two or more instructions and, therefore, satisfies the claim language.

Conclusion

31. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory

period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

32. The following is text cited from 37 CFR 1.111(c): In amending in reply to a rejection of claims in an application or patent under reexamination, the applicant or patent owner must clearly point out the patentable novelty which he or she thinks the claims present in view of the state of the art disclosed by the references cited or the objections made. The applicant or patent owner must also show how the amendments avoid such references or objections.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Benjamin P. Geib whose telephone number is (571) 272-8628. The examiner can normally be reached on Mon-Fri 8:30am-5:00pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Fritz Fleming can be reached on (571) 272-4145. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Benjamin P Geib
Examiner
Art Unit 2181

FRITZ FLEMING
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100
9/17/2006